

# NNexus API

James Gardner

March 1, 2007

**This document outlines the Application Programmer Interface of NNexus.**

NNexus (Noosphere Networked Entry eXtension and Unification System) is an abstraction and generalization of the automatic linking component of the Noosphere system, which is the platform of PlanetMath (planetmath.org), PlanetPhysics (planetphysics.org), and other Noosphere sites.

Users of NNexus will apply the following basic functionality to their corpus. When an entry is rendered for display (either at display time or during offline batch processing), the text is broken down into tokens and scanned for words that invoke concepts that have been defined already (in other entries). These words (or word tuples) are ultimately turned into hyperlinks to the corresponding entries in the output rendering. In addition, when the concepts are added to the collection (or the set of concept labels otherwise changes), entries containing potential invocation of these concept labels are *invalidated* using a special inverted index called the *invalidation index*. This forces these entries to go through link analysis themselves by or before the next time they are displayed.

This automatic system almost completely frees the author from having to “think about links.” It addresses the problems of both outgoing and incoming links, with respect to a new entry or new concepts.

## 1 API

This section describes how to communicate with the NNexus server.

All request to NNexus need to be sent using the following XML form:

```
<request>
...
</request>
```

where ... is any of the following requests. Note: The request must begin with a request tag followed by a newline and the end request must have a newline before and after.

### 1.1 Adding New Link Concepts

In order to add a new concept and link you may send:

```
<addobject>
<entry>
<title>same as above</title>
<defines>thing</defines>
<defines>widget</defines>
<synonym>term3</synonym>
<synonym>phrase of terms</synonym>
<domain>planetmath.org</domain>
<body>The body text</body>
<objid>a3db</objid>
```

```

<linkpolicy>permit 03A</linkpolicy>
<author>1</author>
<class>012A</class>
<class>02ADD</class>
</entry>
<entry>
... Same form as above.
</entry>
</addobject>

```

Notice that it is possible to add multiple objects with one request.

After adding (modifying) an entry to (in) Nnexus the system will return a list of object ids that need to be invalidated in the following form:

```

<response>
<invalid>ExternalID</invalid>
<invalid>AnotherExternalID</invalid>
</response>

```

We can delete an object by passing

```

<deleteobject>
<objid>1232</objid>
<domain>wikipedia.org</domain>
</deleteobject>

```

## 1.2 Getting invalid objects

A user may also request all invalidated objects by sending

```
<getinvalidobjects/>
```

and the server will return all invalidated objects as

```

<update>
<invalidate>
<objid>12</objid>
<objid>123</objid>
<objid>145</objid>
<objid>1125</objid>
</invalidate>
</update>

```

## 1.3 Obtaining Links for New Articles or Terms

There will be three schemes for obtaining links from Nnexus. The first two will use the entire article and the last will only find links for terms. Note: Only the Simple Mode is currently supported.

### 1.3.1 Simple Mode

```

<article> full text of article </article>
<class>03FA2</class>

```

this will return

```

<response>
<body>
full text of article with links added.
</body>
<links>
a list of the links in http://form separated by commas.
</links>
</response>

```

### 1.3.2 Detailed Mode

```

<detailedreq>
<article> full text of article </article>
<class>03FA2</class>
</detailedreq>

```

this will return

```

<detailedresp>
<article> full text of article </article>
<entry>
<place>12,13</place>
<option>
<link>link.to/article</link>
<domain>domain.com</domain>
</option>
<option>
<link>link.to/otherarticle</link>
<domain>otherdomain.com</domain>
</option>
</entry>
<entry>
<place>12,13</place>
<option>
<link>link.to/article</link>
<domain>domain.com</domain>
</option>
</entry>
</detailedresp>

```

### 1.3.3 Interactive Mode

```

<req>
<term>term or phrase</term>
<class>03FA2</class>
<dclass>desired classification of article to link to</dclass>
</req>

```

Note: dclass is optional After sending this information the server will return

```

<resp>
<link>http://link.to/the_article</link>
<linkid>12313</linkid>
</resp>

```

## 1.4 Exporting Concept Labels

To access the concept labels that Nnexus is currently using you may pass

```
<getconcepts>
<domain>domain.org</domain>
<domain>otherdomain.org</domain>
</getconcepts>
```

and the server will return the xml config files as

```
<config>
<domain>domain.org</domain>
<contents>contents of domain.org config file</contents>
</config>
<config>
<domain>otherdomain.org</domain>
<contents>contents of otherdomain.org config file</contents>
</config>
```

## 2 Examples

Contact James Gardner (pebbler@gmail.com) for examples and help.